# Ministério da Educação

# UTFPR - Universidade Tecnológica Federal do Paraná - Câmpus Pato Branco

# Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

# Departamento Acadêmico de Informática

**Proposta de Projeto para o Trabalho de Conclusão de Curso**

**INDEXAÇÃO DE DADOS MÉTRICOS EM BANCOS DE DADOS COMPLEXOS**

Acadêmico: Leandro Menegazzo Franceschetto

Orientador: Dr. Ives Renˆe Venturini Pola

Co-Orientadora: Dra. Fernanda Paula Barbosa Pola

1. **INTRODUÇÃO**

Os sistemas computacionais atuais precisam manipular tipos de dados complexos, tais como: imagens, vídeo, séries (sequências) temporais, entre outros. Porém, os sistemas de gerenciamento de dados não possuem ferramental completo e apropriado para tanto. Desse modo, a comunidade de bases de dados tem contribuído com o desenvolvimento de novos métodos e algoritmos que permitem manipular tais tipos de dados complexos, de um modo eficiente e completo. Mas a concepção de tais algoritmos demanda um estudo aprofundado das restrições de domínio e de sua integração. O presente trabalho vem nessa direção, no desenvolvimento de algoritmos eficientes para o processamento de consulta a dados complexos.

Busca por similaridade é a abstração mais frequente para comparar dados complexos, e é baseada nos conceitos de proximidade para representar a similaridade embasada em conceitos matemáticos dos espaços métricos [Faloutsos, 1997].

O cálculo de dissimilaridade usa uma função de distância que compara por similaridade os dados. Formalmente, um espaço métrico é um par (S, d), onde S é o domínio dos dados e   
d : S × S → R+ é uma função de distância, ou métrica, que atende às seguintes propriedades para qualquer s1, s2, s3 ∈ S:

• Identidade (d(s1, s2 ) = 0 → s1 = s2);

• Simetria (d(s1, s2 ) = d(s2, s1 ));

• Nao-negatividade (0 < d(s1, s2 ) < ∞ , s1 = s2) e

• Desigualdade Triangular (d(s1, s2) ≤ d(s1, s3 ) + d(s3, s2)).

Funções que atendam a essas quatro propriedades associadas a um domínio de dados criam o que se denomina um Espaço Métrico. É comum que funções de distância sejam associadas a diversos outros tipos de dados, como por exemplo tipos de dados espaciais, o que habilita que se solicitem também consultas por similaridade sobre esses tipos de dados.

A desigualdade triangular de uma métrica é muito utilizada na maioria dos métodos de busca por similaridade para excluir regiões do espaço métrico para responder consultas por similaridade de modo mais eficiente. Especificamente, apenas uma variação dessa propriedade é utilizada, a chamada limite inferior da distância.

Existem basicamente dois operadores de seleção por similaridade: a **consulta por abrangência** (*range query*: Rq) e a **consulta aos k-vizinhos mais próximos** (*k-nearest neighbor query: kNNq*) [Chakrabarti et al., 2004] [Zhang et al., 2010]. Uma consulta por abrangência Rq(sq ,ξ) recebe como parâmetros um elemento sq do domínio de dados (chamado elemento central ou elemento de referência da consulta) e um limite máximo de dissimilaridade ξ, e como resultado recupera todos os elementos da base de dados que diferem do elemento central da consulta por no máximo a dissimilaridade indicada. Um exemplo de consulta por abrangência sobre um conjunto de imagens é “*Selecione as imagens que diferem da imagem Im1 por no máximo 10 unidades*'', sendo que a unidade de distância depende da definição da função de distância.

O uso de estruturas de indexação para buscas por similaridade é interessante para apoio aos diversas operadores por similaridade, como os operadores binários de junção, e alguns operadores unários para eliminar elementos repetidos por similaridade em um conjunto, além de agilizar operações por similaridade de um modo geral.

A estrutura de indexação métrica MM-tree [Pola et al., 2007] é uma estrutura de dados em memória para indexar dados em segmentos de dados temporários oriundos de junções primitivas nas tabelas de dados, normalmente resultados de subselects em consultas SQL. Ela particiona os dados no espaço de modo que dois pivôs são utilizados como referência em cada nó e divide o espaço em quatro partições. Mas ela não utiliza a propriedade de desigualdade triangular para evitar cálculos de distância desnecessários. Deste modo, este projeto visa implementar uma versão da MM-tree que agregue esta propriedade para agilizar as consultas por similaridade.

1. **OBJETIVOS**

O principal objetivo deste trabalho é desenvolver uma estrutura de dados métrica otimizada para agilizar consultas por similaridade particionando os dados em memória principal, dando suporte ao Gerador de Planos de Acesso Físico de um SGBDR.

Primeiramente, será implementada a estrutura de indexação MM-tree para ter base de comparação com as modificações que a melhoram. Em seguida será implementada outra versão desta estrutura com as propriedades métricas. Como resultado teremos uma estrutura de dados mais eficiente.

A etapa de validação de resultados será feita de modo a comparar as duas implementações a partir de conjuntos de dados já existentes na literatura. Nesta etapa, o objetivo é avaliar o número de cálculos de distância e tempo de consulta para uma quantidade grande de consultas unárias simples.

1. **JUSTIFICATIVA**

O desenvolvimento de estruturas de indexação alimenta o principal motor de um banco de dados, que é o módulo de Gerador de Planos de Acesso Físico de qualquer SGBDR. Com base em seletividade de condições de seleção por similaridade, ele avaliar os melhores índices para ser utilizado naquela consulta SQL. Assim, deve-se ter diversas estruturas disponíveis para este módulo poder ter flexibilidade na geração dos planos de acesso, repassados ao módulo de Executor de consultas SQL, de acordo com o núcleo básico de módulos em um SGBDR.

1. **MATERIAIS E MÉTODOS**

Para o desenvolvimento deste projeto, será utilizado um framework de desenvolvimento de estruturas de indexação para apoiar a implementação e testes de novas estruturas - o Arboretum. O Arboretum é um framework de desenvolvimento criado pelo grupo de bases de dados da USP de São Carlos. Ele provê funcionalidades recorrentes em estruturas de indexação para definir um ambiente justo para comparações de desempenho. A versão mais recente deste framework pode ser encontrada no link: *https://bitbucket.org/gbdi/arboretum*.

O aporte teórico matemático será dado pela co-orientadora Dra. Fernanda Paula Barbosa Pola, na aplicação das propriedades desejadas para o aprimoramento do mecanismo de indexação e busca dos dados indexados.

1. **CRONOGRAMA**

As atividades são divididas nas seguintes etapas no desenvolvimento. O cronograma destas atividades é dado na Tabela 1.

1. Estudo e acompanhamento da literatura;
2. Estudo das propriedades métricas a serem incluídas na MM-tree.
3. Modelagem da estrutura de dados para suportar as propriedades métricas desejadas.
4. Preparo da monografia do TCC1.
5. Apresentação do TCC1.
6. Modelagem da MM-tree dentro do framework Arboretum.
7. Implementação dos índices no framework.
8. Testes para avaliação de desempenho das implementações.
9. Preparo da monografia do TCC2.
10. Apresentação do TCC2.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2017 | | | | | | | | |
| Mar | Abr | Mai | Jun | Ago | Set | Out | Nov | Dez |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 2 | 2 |  |  |  |  |  |  |
|  | 3 | 3 | 3 | 3 |  |  |  |  |
|  |  | 4 | 4 |  |  |  |  |  |
|  |  |  | 5 |  |  |  |  |  |
|  |  | 6 | 6 | 6 | 6 | 6 |  |  |
|  |  |  |  | 7 | 7 | 7 |  |  |
|  |  |  |  |  | 8 | 8 |  |  |
|  |  |  |  |  |  | 9 | 9 | 9 |
|  |  |  |  |  |  |  | 10 | 10 |

Tabela 1: Cronograma de atividades.

1. **CONCLUSÃO**

A finalização do desenvolvimento deste projeto propiciará ao aluno experiência no desenvolvimento de sistemas de acesso específico a dados não tradicionais. Exercitará suas aptidões de programação e estará melhor preparado para desafios encontrados no mercado de trabalho. Além disso, os resultados provenientes deste projeto podem ser publicados em veículos de pesquisa, no qual contribuirá para o currículo do aluno também.

**REFERÊNCIAS**

[Chakrabarti et al., 2004] Chakrabarti, K., Ortega-Binderberger, M., Mehrotra, S., and Porkaew, K. (2004). Evaluating refined queries in top-k retrieval systems. IEEE Transactions on Knowledge and Data Engineering (TKDE), 16(1):256–270.

[Faloutsos, 1997] Faloutsos, C. (1997). Indexing of multimedia data. In Multimedia Databases in Perspective, pages 219–245. Springer Verlag.

[Pola et al., 2007] Pola, I. R. V., Jr., C. T., and Traina, A. J. M. (2007). The MM-tree: A memory-based metric tree without overlap between nodes. In Advances in Databases and Information Systems, ADBIS, volume 4690 of Lecture Notes in Computer Science, pages 157–171, Varna - Bulgary.

[Zhang et al., 2010] Zhang, R., Jagadish, H. V., Dai, B. T., and Ramamohanarao, K. (2010). Optimized algorithms for predictive range and knn queries on moving objects. Information Systems Journal (ISJ), 35(8):911–932.